

## **Amendments to the Claims**

The following Listing of Claims will replace all prior versions and listings of claims in the application.

### **Listing of Claims**

1. (Currently amended) A system for assigning a unique network identifier to each program invoked on a computer, the system comprising:

a computer obtaining a plurality of network identifiers allocated to a user, the computer comprising:

an interface mechanism selecting, from the plurality of network identifiers of the user, a first network identifier of the user for a first program invoked by the user on the computer and selecting a second network identifier of the user, different from the first network identifier, for a second program invoked by the user on the computer, and assigning ~~associating~~ the first network identifier to ~~with~~ the first program and assigning ~~associating~~ the second network identifier to ~~with~~ the second program; and

a network communication interface, in communication with the interface mechanism, transmitting the first network identifier with a network communication of the first program and transmitting the second network identifier with a network communication of the second program.

2. (Currently amended) The system of claim 1, wherein the network identifier comprises one of either of an internet protocol address, a host name, and a loopback address.

3. (Cancelled).

4. (Original) The system of claim 1, wherein one of the first program and the second program comprises a user session hosted by the computer.

5. (Original) The system of claim 1, wherein one of the first program and the second program comprises one of an application isolation environment and an application.
6. (Original) The system of claim 1, wherein the computer obtains at least one of the plurality of network identifiers from a server.
7. (Original) The system of claim 4, wherein the server comprises a Dynamic Host Configuration Protocol server.
8. (Original) The system of claim 1, wherein the computer obtains at least one of the plurality of network identifiers from a storage location.
9. (Original) The system of claim 1, wherein the computer obtains at least one of the plurality of network identifiers from a network identifier generator.
10. (Original) The system of claim 1, wherein the interface mechanism selects the first network identifier for the first program during an establishment of the first program.
11. (Original) The system of claim 1, wherein the interface mechanism selects the second network identifier for the second program during an establishment of the second program.
12. (Original) The system of claim 4, wherein the computer concurrently hosts a first user session and a second user session.
13. (Original) The system of claim 4, wherein the computer hosts a second user session subsequent to the hosting of a first user session.
14. (Original) The system of claim 1, wherein the interface mechanism provides the first network identifier of the first program in response to a name resolution request of the first

program and provides the second network identifier of the second program in response to a name resolution request of the second program.

15. (Original) The system of claim 1, wherein at least one of the plurality of network identifiers is allocated to a user of the computer.

16. (Original) The system of claim 1, wherein the interface mechanism comprises a first TCP stack.

17. (Original) The system of claim 16, wherein the interface mechanism comprises a socket library for communication with the network communication interface.

18. (Cancelled).

19. (Original) The system of claim 17, wherein the interface mechanism binds the first network identifier to the first program for socket communication with the network communication interface.

20. (Original) The system of claim 17, wherein the interface mechanism binds the second network identifier to the second program for socket communication with the network communication interface.

21. (Cancelled).

22. (Original) The system of claim 1, wherein the interface mechanism comprises a network packet-manipulation filter.

23-30. (Cancelled).

31. (Currently amended) A method for assigning a unique network identifier to each program invoked by a computer, the method comprising the steps of:

- (a) obtaining a plurality of network identifiers allocated to an user;
- (b) selecting, from the plurality of network identifiers of the user, a first network identifier of the user for a first program invoked by the user on a computer, and a second network identifier of the user, different from the first network identifier, for a second program invoked by the user on the computer;
- (c) ~~associating~~ assigning the first network identifier ~~to with network communication of the first program and~~ assigning ~~associating~~ the second network identifier ~~to with network communication of the second program;~~ and
- (d) transmitting the first network identifier with a network communication of the first program and transmitting the second network identifier with a network communication of the second program.

32. (Currently amended) The method of claim 31, wherein the network identifier comprises one of either of an internet protocol address, a host name, and a loopback address.

33. (Cancelled).

34. (Original) The method of claim 31, wherein one of the first program and the second program comprises a user session hosted by the computer.

35. (Original) The method of claim 31, wherein one of the first program and the second program comprises one of an application isolation environment and an application.

36. (Original) The method of claim 31, wherein step (a) further comprises obtaining, from a server, at least one of the plurality of network identifiers.

37. (Original) The method of claim 31, wherein step (a) further comprises obtaining, from a Dynamic Host Configuration Protocol server, at least one of the plurality of network identifiers.
38. (Original) The method of claim 31, wherein step (a) further comprises obtaining, by the computer, at least one of the plurality of network identifiers from a storage location.
39. (Original) The method of claim 31, wherein step (a) further comprises obtaining, by the computer, at least one of the plurality of network identifiers from a network identifier generator.
40. (Original) The method of claim 31, wherein step (b) further comprises selecting the first network identifier for the first program during an establishment of the first program.
41. (Original) The method of claim 31, wherein step (b) further comprises selecting the second network identifier for the second program during an establishment of the second program.
42. (Original) The method of claim 34, wherein step (b) further comprises hosting concurrently, by the computer, a first user session and a second user session.
43. (Original) The method of claim 34, wherein step (b) further comprises hosting, by the computer, a second user session subsequent to the hosting of a first user session.
44. (Original) The method of claim 31, wherein step (d) further comprises providing the first network identifier of the first program in response to a name resolution request of the first program and the second network identifier of the second program in response to a name resolution request of the second program.
45. (Original) The method of claim 31, wherein step (a) further comprises allocating at least one of the plurality of network identifiers to a user of the computer.

46. (Original) The method of claim 31, wherein the method further comprises using a first TCP stack for network communication.

47. (Original) The method of claim 31, wherein the method further comprises interfacing with a network communication interface using a socket library.

48. (Cancelled).

49. (Original) The method of claim 47, wherein the method further comprises binding the first network identifier to the first program for network communications using the socket library.

50. (Original) The method of claim 47, wherein the method further comprises binding the second network identifier to the second program for network communications using the socket library.

51. (Cancelled).

52. (Original) The method of claim 31, wherein the method further comprises interfacing with a network communication interface using a network packet-manipulation filter.

53-60. (Cancelled).